Project Report:

Group Member Names: Masood Ali, Ikhtiar Ali, Ilyass, Iqrar Ali.

Roll No: 20Cs006, 20CS050, 20CS074, 20CS076

Submitted To: Sir Jamsher.

Project Name: Service Program

Food-delivery services like Food Panda are now offering grocery delivery services and for grocery they should offer multiple different delivery options depending upon the weight of the pack. Make an inheritance hierarchy to represent various types of grocery deliveries. Use class FG Delivery (Super base class) as the base class, then include classes Half Hour FP Delivery and One Day FP Delivery that derive from FP Delivery. Base class FP Delivery should have name, address of the shop & recipient, along with weight of grocery (in pounds) and cost per pound for the delivery. There should be zero argument (default) and multiple argument constructors.

There must be a public member function Get Cost that should return the cost of grocery delivery. Get Cost function should determine the cost by multiplying the weight with the cost per pound.

Derived class One Day FP Delivery is to be inherited from FP Delivery. It should have a data member of fixed fee that the company charges for one-day grocery delivery. One Day FP Delivery should have zero & multiple argument constructors. This class should override parent class member function Get Cost so that it calculates the cost by adding a fixed fee to the cost calculated in base class's Get Cost function. Class Half Hour FP Delivery should privately inherit from class FP Delivery and contain a data member of additional fee per pound charged for half hour delivery service. Half Hour FP Delivery should overload parent function Get Cost so that it adds the additional fee per pound to the cost calculated by parent.

Code:

```
//package food.delivery;

import java.util.Scanner;


 class FoodDelivery {



public static void main(String[] args)

      {

     Scanner cin = new Scanner(System.in);

System.out.println("Enter Name of Resturant");

String n = cin.nextLine();

System.out.println("Enter Your Address");

String address= cin.nextLine();

System.out.println("Enter Your Name");

String rep = cin.nextLine();

System.out.println("Enter Your Weight");

int w = cin.nextInt();


OneDayFGDelivery obj3 = new OneDayFGDelivery(n,address,rep,w);

HalfHourFGDelivery obj2 = new HalfHourFGDelivery(n,address,rep,w);

FGDelivery obj1 = new FGDelivery(n,address,rep,w);


System.out.println("Enter 1(Total cost) 2(Total cost on Half Hour Delivery) 3(Total
cost on One Day Delivery)");

int Chose= cin.nextInt();

 while (Chose<1 || Chose>3)
```

```
 {
System.out.println("Enter 1(Total cost) 2(Total cost on Half Hour Delivery) 3(Total cost on One Day Delivery)");
        Chose=cin.nextInt();
 }

if (Chose==1)
{
int Total = obj1.GetCost();
System.out.println("Total = " + Total);
}
if(Chose==2)
{
int Total = obj2.GetCost();
System.out.println("Total = " + Total);
}
else
{
int Total = obj3.GetCost();
System.out.println("Total = " + Total);
 }
}
}
class FGDelivery
{
   String Name="Resturant";
   String Address="Auto ban";
```

```
    String Recipient="X";
    int Weight=0;
    int per_pound=2;

    FGDelivery()
    {
    }

    FGDelivery(String Name,String Address,String Recipient,int W)
    {
        this.Name=Name;
        this.Address=Address;
        this.Recipient=Recipient;
        this.Weight=W;

    }

    public int GetCost ()
    {
        int Total = Weight * per_pound;
        return Total;

}
}

class HalfHourFGDelivery extends FGDelivery
{
```

```
    int HalfHourCharge=10;
   HalfHourFGDelivery()
   {
   }

   HalfHourFGDelivery(String Name,String Address,String Recipient,int W)
   {
      super.Name=Name;
      super.Address=Address;
      super.Recipient=Recipient;
      this.Weight=W;

   }
   public int GetCost ()
   {

      int Total = (Weight * per_pound) + HalfHourCharge;
      return Total;
   }
}


class OneDayFGDelivery extends FGDelivery
{
   int OneDayCharge=10;

   OneDayFGDelivery()
```

```
    {
    }

    OneDayFGDelivery(String Name,String Address,String Recipient,int W)
    {
        this.Name=Name;
        this.Address=Address;
        this.Recipient=Recipient;
        this.Weight=W;

    }

    public int GetCost ()
    {

        int Total = (Weight * per_pound) + OneDayCharge;
        return Total;
    }
}
```

Output: